Computers and Structures 136 (2014) 56–63

Contents lists available at ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

A constraint satisfaction programming approach for computing manufacturable stacking sequences

S. Zein [a,*], P. Basso [b], S. Grihon [c]

[a] LMS Samtech, 8 rue des chasseurs ardennais, 4031 Angleur, Belgium
[b] Airbus Germany, Kreetslag 10, 21129 Hamburg, Germany
[c] Airbus France, 1 Rond-point Maurice Bellonte, 31700 Blagnac, France

ARTICLE INFO

Article history:
Received 22 October 2013
Accepted 14 January 2014
Available online 18 February 2014

Keywords:
Constraint satisfaction programming
Backtracking
Manufacturable composite structures

ABSTRACT

An algorithm is proposed to generate stacking sequences which comply with the requirements of the composite manufacturers. These rules are the blending and the design rules. The novelty of the proposed algorithm is that it can handle a general blending scheme, where a stacking sequence can be blended with other stacking sequences and it can also be the base of others. This algorithm can have two purposes: generating a manufacturable structure given the results of a preliminary design or defining a design space of a composite structure in a design process.

1. Introduction

Composite structures have a growing importance in the aeronautical and automotive domains due to the weight reduction and the strengthening that they can exhibit. A composite structure can have, for example, a set of panels (like in Fig. 1) and each panel can have its own stacking sequence. The fiber orientation in each ply is one of these four conventional values: $\{-45, 0, 45, 90\}$. The design of the stacking sequences must be such that the responses of the structure to a set of load cases do not violate some safety criterion. Moreover, the stacking sequences must be designed at the computer level such that they meet the requirements of the composite manufacturers. Otherwise, the structure cannot be manufactured.

The manufacturing rules are the design and the blending rules. The design rules define the sequence layout. They can be like, for example, having a certain number of plies per orientation, being symmetric, starting with a ±45 ply. Many papers have addressed the problem of satisfying these rules. However, these rules have been considered as the constraints of an optimization problem where the objective function is the buckling load. In [1–3], the genetic algorithms are used with the penalty method in order to satisfy these rules. In [4–6], the topology approach is used where the design rules are formulated as a penalty function of four real decision variables per ply. These two approaches, based on the penalty method, have the drawback of being unable to satisfy all the design rules at the same time due to the combinatorial nature of these constraints. In [7], the integer programming approach is used where the design rules are formulated using four binary decision variables per ply. Here, all the design rules are satisfied, but this approach is only applicable to linear objective functions.

The blending rule consists in the following. Two adjacent panels must have their stacking sequences such that one is a subset of the other. In [8–11], the stacking sequence guide is used to ensure the blending between panels. The sequence of a panel is a subsequence of the stacking sequence guide. A subsequence of n plies must be the first or the last n plies of the stacking sequence guide. This assumption constitutes the limitation of the method. In [12–15], a shared layer approach is used to ensure the blending. In a first step, the sequences of the panels are optimized without the blending constraint. Then, in a second step, the sequences are rearranged to find a blended structure. This second step is the drawback of the method because it does not take into account the objective function (the buckling) of the initial step. In [16,17], a general definition for the blending is considered without any assumption on the ply drop-offs between the panels. A penalty function, based on the differences (the edit distance) between the sequences of two adjacent panels, is used in the optimization process. This approach showed that is it not efficient when it is coupled with the design rules.

In summary, the previous research has addressed the manufacturing rules using the penalty method in an optimization context. It is not an adequate approach given the combinatorial nature of all the rules. This approach makes a trade-off between satisfying the

---

* Corresponding author. Tel.: +32 492700759.
E-mail addresses: samih.zein@lmsintl.com (S. Zein), pasquale.basso@airbus.com (P. Basso), stephane.grihon@airbus.com (S. Grihon).

http://dx.doi.org/10.1016/j.compstruc.2014.01.016
0045-7949/

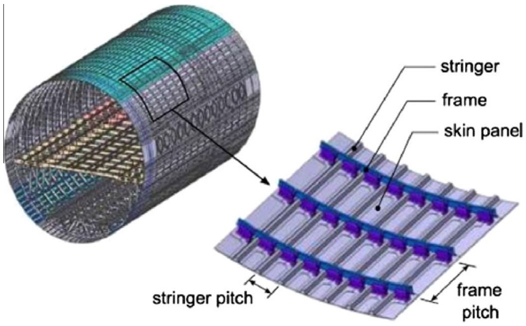

Fig. 1. A composite structure which is a part of a fuselage.

constraints and the objective function. Therefore, it does not guarantee the satisfaction of all the rules.

This paper proposes an algorithm which generates stacking sequences which comply with the blending and design rules. It can handle a blending scheme where the stacking sequence can be blended with other stacking sequences and it can also be the base of others. The advantage of this algorithm is its efficiency in satisfying all the design and blending rules, and thus generating a completely manufacturable structure. This algorithm is only dedicated to the generation of one or many stacking sequences satisfying the manufacturing rules. However it does not give compute the number of admissible stacking sequences. The algorithm does not deal with optimization. In [18] the authors have proposed a combinatorial method to optimize a buckling load based on this algorithm but for a special blending scheme (the first one in numerical experiments): one thickness is associated to one stacking sequence. This work is compared with the topology optimization of the paper [6,19].

The general blending scheme considered in this paper is the case of many industrial applications. It also provides a catalog of stacking sequences which meets the requirements of the composite manufacturers. This catalog can be considered as the design space of a composite structure in a design process. Therefore, this paper does not focus on the mechanical response or the finite elements analysis related to a composite structure. It only concerns the combinatorial algorithm generating manufacturable stacking sequences which are the input of finite elements analysis.

2. Definition of the manufacturing rules

2.1. The blending rule

The blending rule is the following. Let *A* and *B* be two adjacent stacking sequences such that the thickness of *A* is higher than the thickness of *B*. If *A* and *B* are blended, the plies of *B* are a subset of *A*. To illustrate this, consider the composite structure which is a part of the fuselage of an aircraft in Fig. 1. This structure is composed of a set of panels arranged in a grid layout. In this example, a panel can be adjacent to two, three or four other panels depending if its position is at a corner, on a border or inside the structure respectively. In order to reduce the weight of the structure, the panels do not have the same number of plies. Some plies are dropped between two adjacent panels following the blending principle. Fig. 2 is a vertical cut of a structure like the one in the example. One can see the stacking sequences of the panels and the ply drop-offs between them. In this example, there are seven stacking sequences (named A–G). The sequence G is the thickest one, it is composed of the plies $(1,2,3,4,5,6)$. The sequence F has the plies $(1,2,3,4,6)$. Ply 5 is dropped between these two stacking sequences. The sequence B with plies $(1,3,6)$ is blended with two stacking sequences A and C with plies $(1,3,4,6)$. Plies number 1, 3 and 6 are the same in these three sequences, and the two plies numbered 4 are dropped.

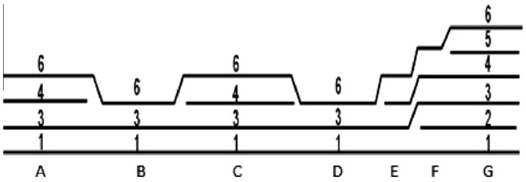

Fig. 2. Blending rule: a vertical cut of a structure showing the ply drop-offs between the panels.

2.2. The design rules

The design rules consist in assigning an orientation to each ply in the structure such that the following rules are satisfied.

- R1: The orientation in each ply must be chosen such that two consecutive plies do not have a gap in the orientation equal to 90°. Thus, $(0,90)$ and $(-45,45)$ cannot be two consecutive plies.
- R2: Maximum four consecutive plies can have the same orientation.
- R3: Symmetric sequences.
- R4: A fixed number of plies of each orientation is defined in each panel. These numbers of plies are the results of a preliminary optimization with the orientation percentages as design variables (see [20,21,13]).
- R5: Uniform distribution of 0 and 90 plies through the sequence: these orientations are not gathered in one part of the sequence. For example the sequence $(0,0,0,0,45,90,-45)$ is not admissible because the zeros are grouped together and they are not uniformly distributed over the sequence.
- R6: A maximum of four consecutive interleaved plies: a maximum of four consecutive plies can be dropped to obtain a subsequence. For example, the two sequences which have the ply numbers $(1,2,3,4,5,6,7,8)$ and $(1,7,8)$ are not admissible because five consecutive plies are dropped (2–5).
- R7: Symmetrical except for odd number of plies in the -45 and 45 directions: a dissymmetry in the center of the laminate is allowed. Asymmetric ± 45 layers in the center of the laminate are separated at maximum by one layer. It is not possible to have a symmetric sequence with an odd number of plies in ± 45, thus a dissymmetry is allowed in the middle of the sequence. For example, consider a sequence with $(3,2,3,2)$ plies of $(-45,0,45,90)$. The only possible way to generate a sequence is like this: $(-45,0,45,90,-45 \mid 45,90,45,0,-45)$. It is a symmetric sequence except in the middle where we have the $-45 \mid 45$ dissymmetry. Another allowed dissymmetry in the middle is $-45,0,45$ and $-45,90,45$. Note that if this rule is considered with the symmetry rule and the fixed number of plies per orientation, the number of 0 or 90 must be odd otherwise the symmetry rule is violated.

3. Graph representation of a composite structure

The constraint satisfaction programming approach is based on building a constraint graph which represents the stacking sequences of a structure together with the manufacturing rules. Consider the set of stacking sequences to be computed. Each stacking sequence is represented with a node. If two stacking sequences are blended together, their nodes are connected with an edge.

The constraint graph can be derived in two cases. The first one is when the stacking sequences need to be computed after the preliminary design of a structure. Fig. 3 shows an example of